Towards the Integration of Value and Coordination Models - Position Paper -

Lianne Bodenstaff [⋆], Manfred Reichert, and Roel Wieringa

Information Systems Group, Department of Computer Science,
University of Twente, The Netherlands
{l.bodenstaff,m.u.reichert,r.j.wieringa}@utwente.nl

Abstract. Cross-organizational collaborations have a high complexity. Modelling these collaborations can be done from different perspectives. For example, the value perspective represents expected value exchanges in a collaboration while the coordination perspective represents the order in which these exchanges occur. How to maintain consistency between different models during design time as well as runtime constitutes a challenging topic. Defining criteria and definitions reflecting the relation between these models during the entire life cycle is not straightforward. Different criteria are used for different models since each model captures a specific aspect of the collaboration. In this paper we investigate the challenges arising when addressing the problem of maintaining adequate and consistent models of a collaboration during the entire life cycle of a collaboration. We propose a framework in which we connect business layer, process layer and implementation layer, presenting the direction for solving this multifaceted problem. We will describe several challenges we anticipate to encounter while implementing our framework.

1 Introduction

The importance of using models to describe collaborations is widely acknowledged [1]. Ensuring consistency between different models is an important and challenging field of research. For example, Nurcan et al. [2] describe the importance of fitting models of organization needs to models of system functionalities. In this paper we focus on adequate design of *collaboration* models, their business processes and support systems. For assessing the collaboration on a management level, business models describing *what* is exchanged between the stakeholders, are developed. These high level models help to clarify expectations of each stakeholder in the collaboration. Moreover, coordination models containing the coordination of business processes, are developed to describe *how* these exchanges are accomplished. These coordination models form the basis for implementing and execution by the information system (IS).

[⋆] Supported by the Netherlands Organisation for Scientific Research (NWO) under contract number 612.063.409

Adequate design of business processes and support systems necessitates precise *consistency* notions for the different aspects of the IS. A necessary condition for adequate design of business processes is ensuring consistency between the different models of the IS. Moreover, consistency between these models and the running IS should be ensured and maintained. In particular, ensuring consistency during design time and during runtime, constitutes a challenging task.

Well maintained business models are crucial for evaluating whether the business goals are met. However, relating high level business models with data from the running IS is not straightforward. Therefore, it is highly important to establish the relation between these different models. When, for example, software decisions are made based on inconsistent models, violation or invalidation of these agreed upon models is a high risk. As a consequence, a collaboration might turn out to be not profitable for one or more stakeholders. When the business processes are not consistent with the available resources or with the expected value exchanges, the business collaboration is not modelled adequately.

Maintaining this consistency at an operational level, through the life cycle of the collaboration, is not straightforward. Our aim is to provide a framework for ensuring consistency between value and coordination layer based on real life data from the running system, capturing the dynamic nature of collaborations. When we succeed in maintaining consistency during design time and runtime we can say that we have achieved *adequate* system design. This position paper discusses challenges arising when relating models through *consistency*. Furthermore, we discuss challenges in *maintaining* adequate and consistent models at an operational level.

This paper is structured as follows: Section 2 explains value and coordination modelling. Section 3 holds the core of this paper and describes challenges maintaining adequate models after which Section 4 describes a first solution approach. We conclude this paper in Section 5.

2 Value and Coordination Modelling

In this section an example of value and coordination modelling is presented by means of a small example. The case represents an insurance company selling an insurance to one of its customers. For collaboration modelling, *value models* are especially important since they represent objects of value exchanged between the stakeholders. It estimates revenue for each stakeholder involved through a method of cost-benefit analysis (like e.g. Net Present Value (NPV) [3] and Return on Investment [4]). Figure 1 depicts a value model of our example case. Here, e^3-value [5] modelling technique is used for illustration but other techniques like e.g. REA [6] and Business Modelling Ontology [7] are available. Each stakeholder is represented as an *actor*. Two *value transfers* are present. One value object, *premium*, is transferred from the customer to the insurance company. The other value object, the *insurance* itself, is transferred from the insurance company to the customer. The consumer need is "having health insurance for one year". This is represented by placing the *start stimulus* at the customer. Now, the set of *value objects* that needs to be transferred to fulfill the consumer need, consists of all value transfers connected through the *dependency path* in the model. The *explosion element* with ratio 1 : 12 indicates twelve payments in a year and the *estimated average* of each payment is 80 Euros. Figure 2 denotes the realized log entries after one year.

A *coordination model* can be evaluated as a *global process model.* The public coordination process *between* stakeholders is strictly separated from internal business processes of stakeholders for confidentiality. *Ordering* of tasks and exchanges is referred to as *execution sequences* in the coordination model. This ordering is not present in the value model. Coordination model examples are e.g. Finite State Automata, Petri Nets [8], Statecharts and activity diagrams. Figure 3 depicts the coordination model of our example case as a Petri Net [8]. The static part of the Petri Net consists of places (indicated as circles) and transitions (indicated as rectangles) which are connected with each other through arcs. Message exchanges are represented as places and tasks are represented as transitions in the coordination model represented as a Petri Net. Figure 3 depicts the message exchange of paying the *premium* by the customer to the insurance company. The value exchange of the *insurance* depicted in the value model in Figure 1 is not depicted as a message exchange in the coordination model, simply because there is no message exchanged representing this value object.

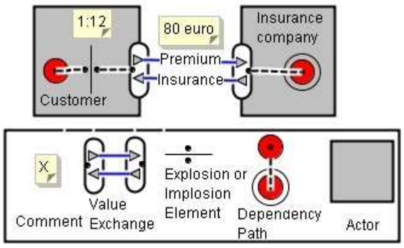
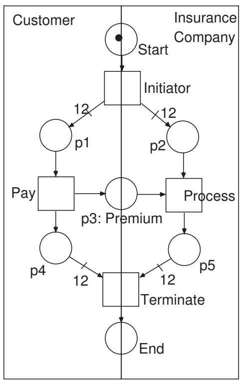


Fig. 1. Value Model

Premium Paid	Amount
8 times	80 Euro
3 times	70 Euro

Fig. 2. Log File Data



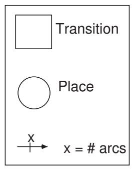


Fig. 3. Coordination Model

3 Using Consistency in Maintaining Adequate Models: Challenges

Establishing the relation between value model and coordination model during runtime as well as during design time is not straightforward. Figure 4 depicts the different relations and their nature between these constructs.

The first relation is between value and coordination model during *design time*. Value model and coordination model should be consistent during design time, e.g. they should model the same system. Consistency checking during design time disregards the dynamic aspects of the collaboration and is therefore referred to as *static consistency*. In [9] an intuitive definition of consistency between a value and a coordination model has been defined which we adapt for our static consistency checking. Here, consistency is checked by matching value exchanges in the value model with message exchanges in the coordination model so that these value exchanges are accomplished and vice versa. Two models are now considered to be consistent if for every set of value exchanges there exists an execution sequence in the coordination model However, this definition only ensures consistency during design time, checking whether value transfers and message exchanges are possible. Before implementation, consistency should also be checked while *simulating* execution of the collaboration. Checking consistency during simulation is referred to as *semi-dynamic consistency* checking and is depicted in Figure 4 as the second part of the relation between value and coordination model.

Another relation in Figure 4 is between coordination model and IS. The coordination model is *implemented and executed* by the IS. Much research in Software Engineering community and Workflow community focusses on this relation. During *runtime*, i.e., at an operational level, data concerning message exchanges between stakeholders is stored in log files. These log files *reflect the behavior* of the IS. Checking consistency between coordination and value model during runtime, means comparing this data, representing the execution of the coordination model, with value exchanges in the value model. In the value model estimations are made on the number of occurrences of value exchanges and on the average value of a value exchange. Checking consistency at an operational level between value model and coordination model is referred to as *dynamic consistency* checking. This is achieved by *monitoring consistency* between value model and log files. A last relation in Figure 4, out of the scope of this paper, is between data in the log files and coordination model. Work like e.g. [10] focusses on this relation by *process discovery* and checking *compliance* of the coordination model with data.

Next, we present interesting challenges in maintaining consistency during *design time*, in maintaining consistency during *runtime* and in consistency checking with *real life scenarios*. Specific problems arising when addressing these three main topics are discussed.

Problem 1. How to ensure consistency between value and coordination model during *design time*?

When developing adequate models describing a collaboration, they have to be consistent with each other, i.e., each model should implement the same system. An important and challenging first question is how to ensure consistency between these models. In this stage we focus on static consistency between models, i.e., consistency between models during design time of the collaboration. We adopt

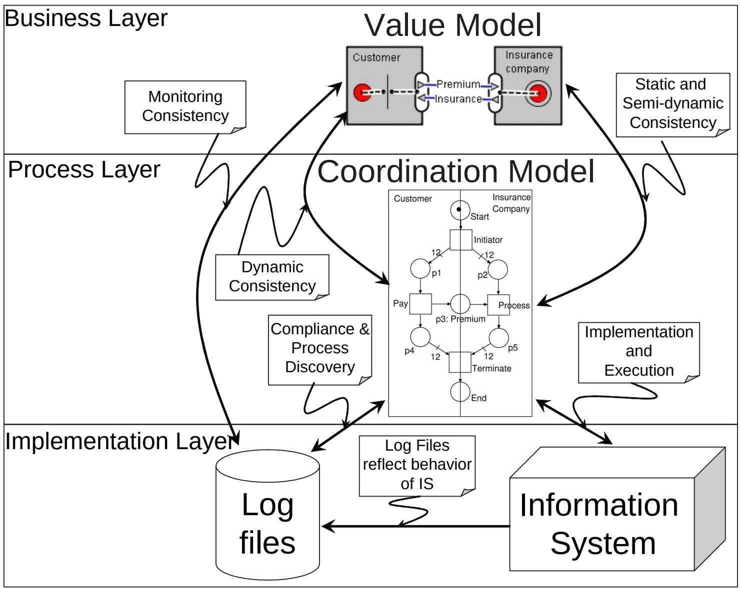


Fig. 4. Overview Model

the intuitive definition given in [9] as discussed in the beginning of this section. This definition focusses on the static aspects only, i.e., two models are considered consistent when there *exists* a possible execution of the models, satisfying sets of value exchanges and execution sequences. However, the *existence* of such an execution is not sufficient. It should be checked whether by implementing this system these exchanges are not only *possible* but indeed *executed*. One method for checking these dynamic aspects *before* implementation is running a *simulation* of the models, checking *semi-dynamic* consistency. Furthermore, we so far assumed every value transfer in a value model can be mapped to a message exchange in the coordination model and vice versa. This is, however, not necessarily the case. How do we handle situations where value transfers in the value model are not relatable to message exchanges in the coordination model and vice versa?

Problem 2. How to ensure consistency between value and coordination model during *runtime*?

An important research challenge is to ensure consistency of models after implementation of the collaboration. Now, models should be consistent with the running implementation. Inconsistencies occur when the system behaves different from what is expected by data in the models. We propose to gather data from the *log files* produced by the IS of the stakeholders in the collaboration and compare this data with the expected *value exchanges* in the value model. This data represents the implemented coordination model. For example, when it was agreed by the stakeholders to exchange over a certain period of time x number of insurances for y amount of money per insurance, then the log data should show *message exchanges* containing these numbers and amounts between the stakeholders over that period of time. This data is modelled in the coordination model. By *monitoring* the collaboration during runtime, continuous assessment of the models is achieved. It should be noted that monitoring the system based on log files only accomplishes monitoring of these value exchanges which are actually captured in the log files. For example, the value exchange *insurance* in Figure 1 does not have a matching entry in the log file since it is a provided service and not a money or product entity.

Another important question is how to address inconsistencies arising during runtime? When, while monitoring the collaboration, inconsistencies occur, several approaches are possible. First, one or more *models* can be adapted according to the data from the log files. In this case the models *describe* the implemented system and are updated according to real life data. The models now can be used for evaluating functioning of the collaboration on a management level. Second, the *implementation* of the IS can be adapted. In which case the models provide a *prescription* on how the collaboration should function. The goal of maintaining models and implementation is to ensure consistent and adequate models and business processes. However, making structural changes in one model, attempting to achieve consistency with data, can cause inconsistencies with other models. This gives rise to the question of how to ensure consistency while continuously adapting models and implementation? Dynamically and continuously adapting models complicates ensuring consistency. *Management of changes*, e.g. the ADEPT project [11] is an important field of research in business process design, addressing these challenges.

Problem 3. Which real life scenarios increase complexity for ensuring consistency?

In the problem statements above we assumed a "perfect" world, e.g. models are complete, gathered data is error free and without noise. However, real life scenarios can give rise to increasing complexity for ensuring consistency and adequate design. Here, we identify scenarios influencing consistency checking. For example, how to handle incomplete or erroneous models and data? Using data for automated consistency checking, implies a necessity for complete and noise-free data. A challenging task is to ensure complete and noise-free data. Moreover, business models of a collaboration can be incomplete or non-existing. Is it, for example, possible to construct a value model by using log file data from the running system? Much research has been done on *process discovery*, e.g. by van der Aalst et al. [10]. However, these problems have not been addressed for value modelling.

Another challenge is *availability* of data. How is necessary data harvested from log files? This challenging question is multifaceted. A collaboration consists of several stakeholders, each having their own IS. Consequently, parts of the log files are provided by different stakeholders with different IS. These different log files might be contradicting or inaccessible due to confidentiality. The value model, however, is a global, agreed upon model. Consistency checking for the complete value model implies the necessity for one merged log file, contradiction free and complete. Gathering this data constitutes a challenging task.

Another interesting challenge is handling *complex* business models or consumer needs. Season dependent behavior, for example, gives rise to complexity in continuous consistency checking. In the value model, an expected *average* over a period of time is calculated. When, for example, the expected amount of ice-skates sold in a year is calculated, the log files will see a peek of sales during winter while during summer less ice-skates will be sold. Now, the collaboration is continuously monitored based on the average in the value model and will give large deviations in winter as well as in summer. Another problem in complex models is *non-matching value transfers and message exchanges*. As a last challenge we mention *scalability*. How to maintain adequate models, representing large collaborations in a dynamic environment? An interesting question is how well solutions on monitoring collaborations on a *global* level scale.

4 First Solution Approach

As a first solution approach to the research challenges described in this paper, we have investigated issues in consistency checking between value and coordination models [12]. Moreover, a dynamic consistency relation has been defined and we developed an approach to achieve dynamic consistency through comparing the value model according to results provided by log files [13]. In this approach we first check *static consistency* by matching value transfers of money or products with message exchanges in the coordination model. For example, money value transfer *Premium* in Figure 1 matches with message exchange *Premium* in Figure 3. After implementation we match real life data with value transfers in the value model. For example, the estimated twelve payments of premium in Figure 1 are matched with the realized 11 transfers of log data in Figure 2, as well as the estimated average value of 80 Euros with the realized average of $77,25$ Euros. Here, log data is not consistent with estimations in the value model. Part of this research is a *proof of concept* implementation. However, this research does not address *complex situations* as described in this paper.

For the near future we plan to *validate* our approach by extensive testing through artificially created log files and later on also through real life data gathered from case studies. Furthermore, the research challenges described in this paper will be systematically addressed, extending our current research.

5 Summary

In this paper we describe important research topics concerning adequate system design during *design time* as well as *runtime*. More specifically, we address research challenges concerning *static* as well as *dynamic consistency* checking between business and coordination models. For us, adequate design means continuous *monitoring* of different models, ensuring their consistency. One way of monitoring these models is checking their compliance with the real life instances as we get them from the log files. This is what we propose in this paper. Therefore adequate design does not mean considering just adequate modelling of business processes but also of adequate modelling of the information support system. For example, it should be modelled which value exchanges are to be accomplished by implementing the business processes.

References

1. Barrios, J., Nurcan, S.: Model driven architectures for enterprise information systems. In: Advanced Information Systems Engineering. Volume 3084., Springer Verlag (2004)
2. Nurcan, S., Edme, M.H.: Intention-driven modelling for flexible workflow applications. Software Process: Improvement and Practice **10**(4) (2005)
3. Laudon, K., Laudon, J.: Essentials of Management Information Systems. 5 edn. Prentice Hall (2003)
4. Friedlob, G.T., Plewa Jr., F.J.: Understanding Return on Investment. John Wiley & Sons, Inc. (1996)
5. Gordijn, J., Akkermans, J.M.: Value-based requirements engineering: Exploring innovative e-commerce ideas. Requirements Engineering **8**(2) (2003) 114–134
6. McCarthy, W.E.: The rea accounting model: a generalized framework for accounting systems in a shared data environment. In: Accounting Review. Volume 57. (1982) 554–578
7. Osterwalder, A., Pigneur, Y.: An e-business model ontology for modeling e-business. In: Proceedings of the 15th Bled E-Commerce Conference - Constructing the eEconomy. (2002)
8. Peterson, J.L.: Petri Net Theory and the Modeling of Systems. Prentice-Hall (1981)
9. Zlatev, Z., Wombacher, A.: Consistency between e^3-value models and activity diagrams in a multi-perspective development method. In: OTM Conferences (1). (2005) 520–538
10. van der Aalst, W.M.P., van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, A.J.M.M.: Workflow mining: A survey of issues and approaches. Data & Knowledge Engineering **47**(2) (2003) 237–267
11. Reichert, M., Dadam, P.: ADEPT_{flex}-Supporting dynamic changes of workflows without losing control. Journal of Intelligent Information Systems **10**(2) (1998) 93–129
12. Bodenstaff, L., Wombacher, A., Reichert, M.: Dynamic consistency between value and coordination models - research issues. In: Proceedings of the International Workshop on Modeling Inter-Organizational Systems (MIOS-CIAO'06). Lecture Notes in Computer Science 4277, Springer Verlag (2006) 802–812
13. Bodenstaff, L., Wombacher, A., Reichert, M., Wieringa, R.: Monitoring collaboration from a value perspective. In: Proceedings of 2007 Inaugural IEEE International Conference on Digital Ecosystems and Technologies IEEE-DEST 2007. (2007) 134–140